Multi-Robot Cooperative System For Object Detection

Duaa Abdel-Fattah Mehiar
AL-Khawarizmi international collage
Duaa.mehiar@kawarizmi.com

Abstract- **The present study proposes a multi-agent system based mobile robot which can detect objects in a structured environment based on a set of given object features. The multi-agent robotics system consists of number of identical robots of NXT Mindstrorm type. Each robot is controlled by appropriate agent-based software according to its mission. The robot hardware and software were developed to fit the object detection mission. The proposed multi-robot system is superior enough to do the following: navigate around to locate objects, communicate with other agents (i.e. robots), pick-up objects, and move them back to the home position. We used multi-agent technology in both communication and program design. The SPP protocol for connecting the NXT mobile robot systems and the NXT-G programming language were adopted. To explore the advantages of the proposed system we build a test environment to check the multi-agent system behavior. The result is promising.**

I. Introduction

Agent technology has been developed rapidly over the last few years and is especially used when solving problems in a world that changes frequently. Multi-agent robotics systems consist of number of identical robots, each controlled by identical agent-based software. A multi-agent system is a system which composed of multiple interacting clever agents. Multi-agent systems can be used to solve problems, which are difficult or impossible to be solved by a single agent. The main idea with agents is that you do not give those orders, but let the individual agent decide what to do according to what they discover in the world. A good way to illustrate the key concepts in a multi-agent system, such as behavior, negotiation and communication is by implementing a robotic multi-agent system. NXT mobile robot is a type of mobile robot which has been recently used to explore various possibilities for agent technology. NXT has been used in various types of application as a model for an intelligent robot [1] [2].

An agent is a computer system situated in some environment, and that is capable of autonomous action in this environment to meet its design objectives. Autonomy is a concept to pin down precisely, it is simply can be defined in the sense that the system should be able to act without the direct intervention of humans or other agents, and should have control over its own actions and internal state.

Multi-agent systems, which mean collections of specialized agents working in parallel, which helped in solving complex problems, not previously solved. Moreover, agent-based simulators have been used to better understand cooperation and competition in society. It is therefore not surprising that agent-based systems have been successful in many varied domains, including Ecommerce, online learning, medicine, entertainment, human-computer interaction, business management and traffic control. In recent years, intelligent agents and multi-agent systems have become a highly active area of AI research.

The cooperative multi-robot search method can be used in a military context, for the searching of buildings for hostiles, booby traps etc. before troops actually enter the buildings. Remotely operated robots are already in use for reconnaissance and surveillance missions. However, searching with a single robot can be time consuming and offer only limited coverage at any instant of time. Operating with multi-robot systems will shorten the search time and increase the instantaneous coverage.

A possible operating concept is to put a remotely operated robot in command of a group of autonomous robots. The cooperative search method can also be applied in an open environment like roads and road junction, to fulfill a similar function of searching for booby traps [5].

II. Related Work

Many researches had been discussed this issue, i.e.

In 2007, Anders Lemke, Johan Laidlaw and Lars Zilmer-Pedersen implemented a system consisting of four Lego Mindstorm NXT robots equipped with light sensors and a touch sensor. They implemented a multi-agent system capable of solving some basic problems. Including implementing the A*-algorithm to find the shortest paths. Finally, they implemented a human controller based system [4].

In 2008, Mohammad Omar Salamah and Alaa Sheta proposed a new proposed landmine detection system based on number of mobile devices. They proposed simple components and assumed a number of assumptions to achieve a good success for that system. In this study, they used only one agent to perform for especial actions learned to such robot. They used small mobile robot, mobile phone and laptop server and camera.

They developed an image processing software to help in detecting the target or object in infrared image [3].

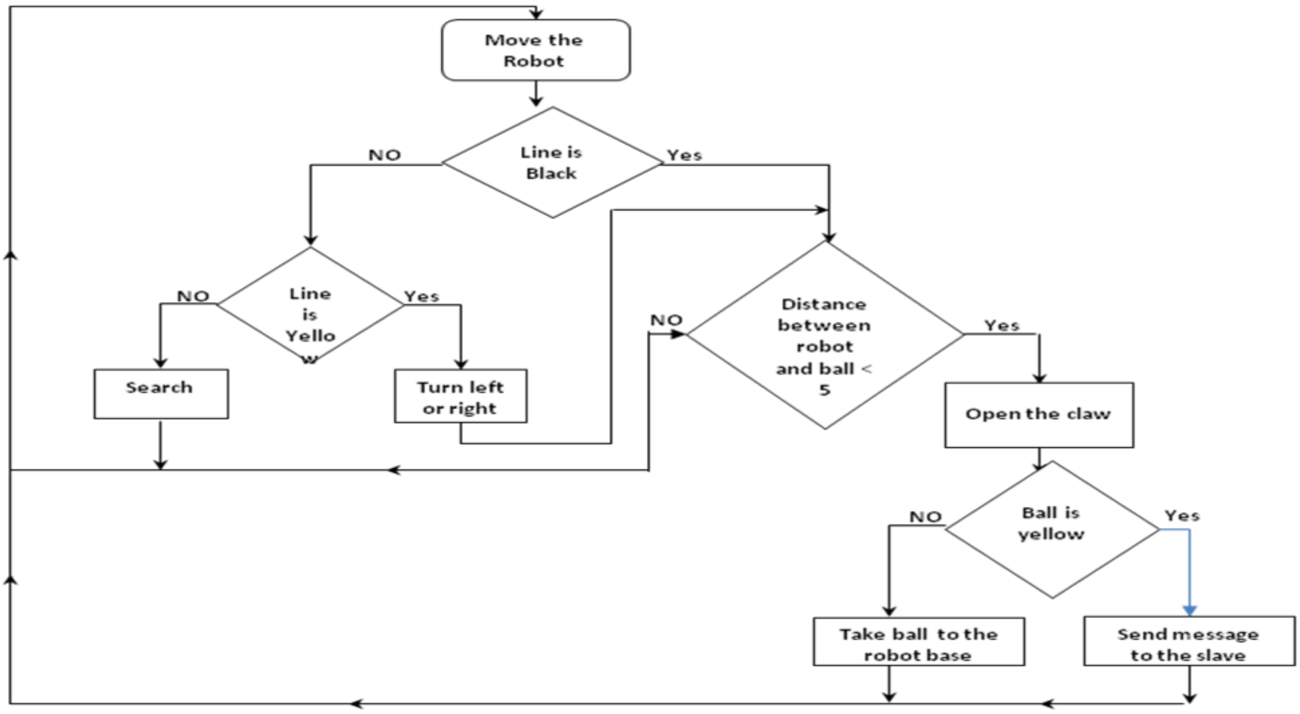

Figure 1: The Activity Diagram of the Double Trained Research System

III. SYSTEM ARCHITECTURE

Our objective is to design a system, which can control mobile robots to explore a path and detect any object located in its path. The main goal of the system is to detect and remove all objects on its path.

A. Double Trained Research System

The methodology of the Double Trained Research System operation can be summarized as follows:

- Case (1): master and slave make search operation if they don't find balls, they send message to each other, "nothing was found".
- Case (2): when master finds the blue ball, it will pick up and take it home, and if the slave found yellow ball, it will pick up this ball and take home.
- Case (3): when master finds yellow ball, it sends message to the slave telling it that a yellow ball is found so the slave will continue searching. The slave robot will do the same thing when it finds a blue ball.

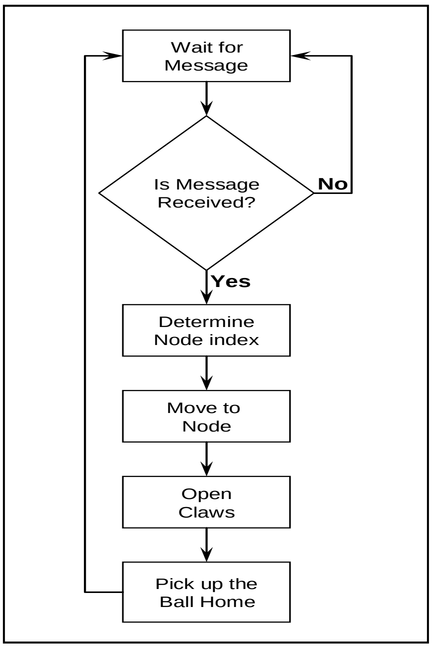

Figure 2: Activity Diagram of the Master Robot

B. Master Guidance Research System

The principle of the Master Guidance Research System operation differs from that of Double Trained Research System, and can be summarized as follows:

- Case (1): master robot makes a search while slave robot is waiting in its location. If master robot does not find any balls it will return to its location and finish searching.
- Case (2): when master finds the blue ball, it will pick up and take it home, and if it finds yellow ball, it will send a message of the ball node number to the slave robot.
- Case (3): When slave robot receives the message, it will go to the node number and pick up the yellow ball and take it home.

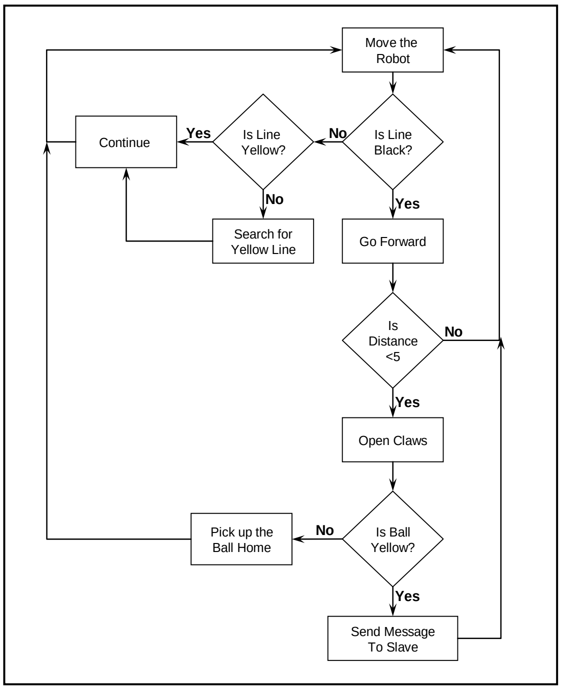

Figure 3: Activity Diagram of the Slave Robot

IV. Navigation Components for the Proposed System

Navigation is the key robotics concept. Without proper navigation techniques, robot would wander aimlessly. The definition of navigation is to manage or direct the course of robot.

The proposed system implements the Robot Orientation method for localization ("Localization involves the question: Where is the robot now ?"), where the Robot Orientation can determine by recording the rotation of the robot from the initial position of the robot. The only information needed for Robot Orientation is the direction and the distance; the LEGO Mindstorm NXT is provided with motors including a rotation sensor, by keeping an accurate record of the wheel rotation, the distance can be determined i.e. from the rotation sensor reading we can calculate the degrees which the robot rotated.

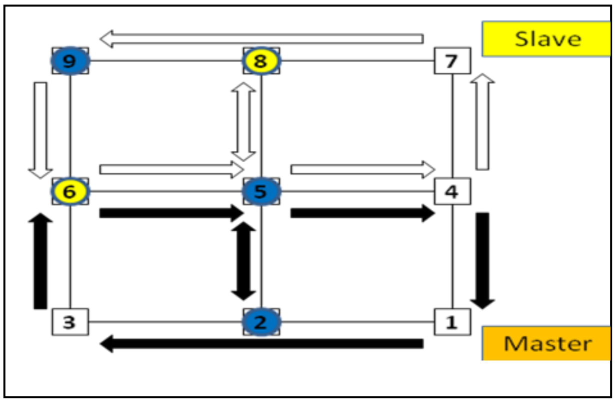

Figure 4 shows a 2 × 2 map. Black arrows show the

Movements of master robot, while white arrows show the movements of slave robot.

V. Communications and Control Software

In this section we describe the software development, functionality and features:

- Communication Software: The communication software will be the bridge between the Master and Slave robots. This feature will expand the range of wireless communication between robots.
- Robot Controller: the robot controller is an NXT-G application developed to control the LEGO NXT via interpreting the messages to the NXT as commands. These messages are exchanged between robots.

A. Communications Software

NXT brick model uses only the synchronize communication type as follows. The NXT brick supports wireless communication using Bluetooth by including CSR BlueCore-4 version 2 chip. The NXT brick can be connected wirelessly to three other devices at the same time but can only communicate with one device at a time. This functionality has been implemented using the Serial Port Profile (SPP), which can be considered as a wireless serial port. The NXT brick can communicate with Bluetooth device that can be programmed to communicate using the LEGO MINDSTORMS NXT communication protocol comment and that support the SPP. It is possible to use wireless communication to send and receive information between bricks during program execution. To reduce the power consumption used by Bluetooth, the technology has been implemented as a Bluetooth class II device, which means that it can communicate up to a distance of approximately 10 meters.

The Bluetooth functionality within the NXT brick is set up as a master/slave communication channel. This means that one NXT within the network needs to function as the master device and that other NXT bricks communicate through it if they need to.

As shown in Figure 5, the master NXT can only communicate with one of the slave devices during a given moment, meaning that if the master NXT is communicating with NXT slave 1 and NXT slave 3 starts sending data to the master NXT, the master NXT will not evaluate the received data until it switches to NXT slave 3.

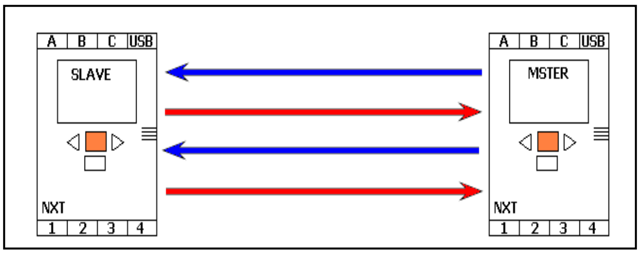

Figure 5: Synchronize Communication between Master and Slave NXTs

An NXT is not able to function as both a master and slave device at the same time because this could cause lost data between NXT devices. This functionality (i.e. serving as a master and slave device at the same time) has been disabled in the standard LEGO MINDSTORMS NXT firmware.

Connections to other Bluetooth devices occur through channels. The NXT has four connection channels used for Bluetooth communication. Channel 0 is always used by slave NXT slave devices in the communicating with the master NXT (i.e. toward the master NXT) while channels 1,2 and 3 are used for outgoing communication from the master device to the slave devices.

B. Control Software

Robot Controller Component is an NXT-G application developed and installed on the NXT. The application will send/receive the request from the other robot, and interpret the request to an execution command.

1) The Robot Controller Component for Double Trained Research System

NXT-G is a special code, which can be used for robots simulation, training, learning and implementation. In order to make the robot performing the needed tasks, this will be carried out through the following steps:

- At the beginning, the program we make the robot move by using two motors for one complete rotation
- The second step: the program makes a check on the path using the light sensor. If detects a black line, then it will go to the next step, either it will search for black line. If it finds an intersection (Detects a yellow node) then the robot will turns right.
- The third step: The robot is making a check using ultrasonic sensor if there is an object or not, if there is an object, the robot goes to the next step, else it will continue searching.
- The fourth step: The robot opens its claws and moves toward the object, then close the claws.
- The fifth step: When the robot griping the ball, it makes a check on its color, if it is blue, it will be put in its home area, and the master then continued its search. If it was a yellow ball, it goes to the sixth step.
- The sixth step: The master robot sends a Bluetooth message for the slave robot telling it to continue searching, and the master robot continue searching for another ball.
- The seventh step: If the slave robot finds a blue ball, it will send a message for the master robot, which receives the message.
- The Slave program is similar to Master program. The difference is in the ball color, such that if the slave finds yellow ball, it will catch it and move it to its home area. If it finds blue ball, it will send a message to the master robot. If the slave robot receives message from the master robot, it will continue searching for a yellow ball until it is found.

2) The Robot Controller Component for Master Guidance Research System

This application will send the request from the master robot to the slave robot, and interpret the request to an execution command.

- At the beginning the program we make the robot move by using two motors for one complete rotation.
- The Next step: the program makes a check on the path using the light sensor. If detects a black line, then it will go to the next step, either it will search for black line. If it finds an intersection (Detects a yellow node) then the robot will turns right.
- The robot is making a check using ultrasonic sensor if there is an object or not, if there is an object, the robot closes its claws and goes to the next step), else it will continue searching.
- When the robot griping the ball, it makes a check on its color, if it is blue, it will be put in its home area, and the master then continues its search. If it was a yellow ball, it goes to the next step.

- The master robot sends a Bluetooth message for the slave robot telling it the position of the detected yellow ball and the master robot continue searching for another. When the master sends data to the slave the message specifies a mailbox, which is a number between 0 and 9 (represented as 1 to 10 in NXT-G programs). For the slave, this builds 10 virtual channel on top of one master slave connection.

- The difference between master robot NXT-G controller and Slave robot NXT-G controller components is that the slave is controlled by the master robot, such that if the master robot finds a yellow ball, it will send its location to the slave robot, otherwise, the slave robot will stay in its home region.

- For example, when master robot finds a yellow ball in node 3, it will send a message to slave robot to mailbox "3". When slave robot receives a message, it will check its mailboxes, until it finds the message. The mailbox index represents the node at which the yellow ball exists. Then, the slave robot will move toward the determined node, and catches the ball to its home region.

VI. EXPERIMENTAL RESULTS

A. Some Problems and Solutions

Through the implementation of the two systems for different area sizes, some problems appeared such as deviation and system fails. I suggested some solutions that solved these problems and enhanced the system performance.

1) Deviation

This problem was solved in two ways, one of them is hardware way, and the other is software. In the hardware, small wheel located in the back used to facilitate the movement and reduces friction. In the software treatment, I used the black lines and light sensors to detect the robot path.

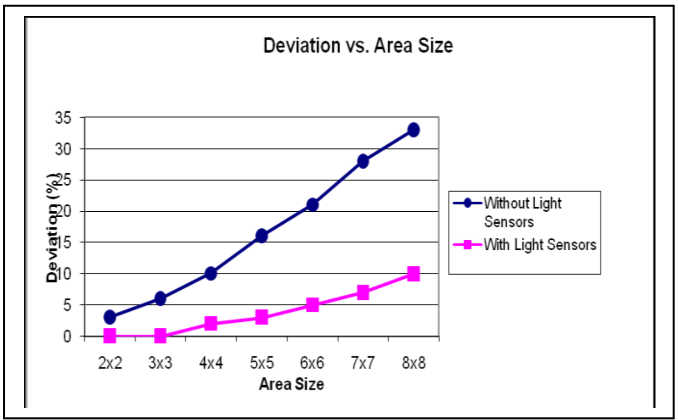

Figure 6: Light Sensor Effect on Robots Deviation

Table 1 shows the deviation percentage with respect to time for different area sizes with and without light sensors, and Figure 6 shows the related curves.

TABLE 1: LIGHT SENSORS EFFECT ON DEVIATION

Area Size	Deviation Without Light Sensors	Deviation With Light Sensors
2×2	3%	0%
3×3	6%	0%
4×4	10%	2%
5×5	16%	3%
6×6	21%	5%
7×7	28%	8%
8×8	33%	10%

2) Time Efficiency:

The time needed to complete a mission in relative to the area size, such that as the area size increases, the time needed to complete the mission increases. Table 2 shows the mission times for different area sizes, and Figure 7 shows the related curves.

TABLE 2: LIGHT SENSORS EFFECT ON DEVIATION

Area Size	Mission Time (second)	
	Double Trained System	Master Guidance System
2×2	16	25
3×3	28	46
4×4	52	85
5×5	102	160
6×6	160	295
7×7	435	620
8×8	760	1060

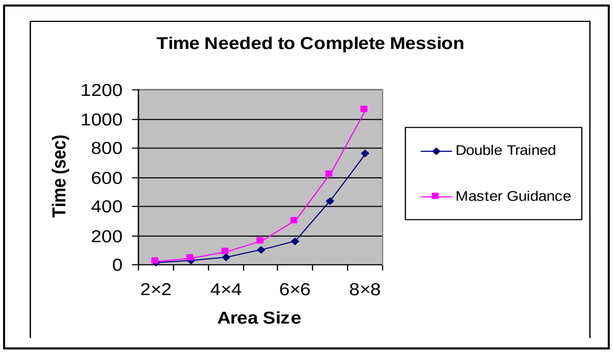

Figure 7: Mission Time vs. Area Size

VII. CONCLUSIONS AND FUTURE WORK

In this paper, we presented our initial idea on developing a multi-agent system based mobile robot for object detection. The system architecture and software were proposed. We tested our system under various operation conditions to make sure that it will succeed the test in real life. The design of the system went through number of stages. They include hardware design, software program development; object recognition system (i.e. feature selection), building a test environment and finally test the system using number of scenarios.

Double Trained Research System has an advantages over Master Guidance Research System in the time needed to complete the mission, this because both robot start searching at the same time in Double Trained system, while in Master Guidance System, only the master starts searching while the slave is waiting for commands.

Another advantage of Double Trained System that if the master robot fails in its mission, the slave robot can complete its mission. On the other hand, in the Master Guidance system, the slave is guided by the master, so if the master fails, then the overall system fails. Due to this advantage, it is preferable to use the Double Trained System in areas or fields where it is very common that any robot can fail such as fire areas, earthquakes, and wars regions.

Also, the Master Guidance System has an advantage of the lower cost, because the master robot has different architecture than slave robot due to the difference in their operation principles.

We plan to enhance the proposed methods for detecting and controlling the agent (i.e. mobile robot) to get better maneuver of the robot. With suitably selected robots, our system can help in search and rescue operation in disaster area without human involvement. This will reduce the time to discovery of the victims and save life's.

REFERENCES

[1] L. M. NXT'reme. LEGO MINDSTORMS NXT Bluetooth protocol, version 1.00 edn.

[2] G. R. Andrews, Foundations of Multithreaded, Parallel, and Distributed Programming. Addison Wesley Longman, Inc, 1998.

[3] Mohammad Omar Salamah, Alaa Sheta, Analysis and Design of a Landmine Detection System Using Mobile Devices, Master degree thesis, Al-Balqa' Applied University, 2008.

[4] NXT Lego the mobilrobot. http://mindstorms.lego.com.

[5] Ghada A. Al-Hudhoud. " Multi Agent Communication Model to Control a Group of Networked Mobile Robots Performing a Team Work", The AI Exchange, Vol. 13,pp. 30-34.,Fall 2002.

[6] Anders Lemke, Johan Laidlaw ,Lars Zilmer-Pedersen, Developing Multi-Agent Lego Robotics. Technical University of Denmark, 2007.

[7] J.G. Blitch, "AI Applications for Robot Assisted Urban Search and Rescue," The AI Exchange, Vol. 9, pp. 15-20., Fall 1995.

[8] Jonathan Dixon and Oliver Henlich "Mobile robot Navigation": Imperial College, London: 10 June 1997

[9] S. Julier and J. Uhlmann, "Simultaneous localization and map building using split covariance intersection," in Proc. Int. Conference of Intelligent Robots, Vol. 11, pp. 32-39. 2004.